THE BEGINNERS' GUIDE TO SYN

cm DVD
ON THE DVD
Hear the synth-tastic sounds from our walkthrough examples

> **Of all the core concepts behind computer-based music production, synthesis and synthesisers are certainly among the most intimidating 'on paper'. This is in large part due to the amount of fairly esoteric jargon that remains in place, but once you come to terms with all of that, synths aren't anywhere near as complicated as they first seem.**

In this feature, we'll guide you through the world of sound synthesis and the components that form the 'engine' of most software synthesisers, as well as showing you how to program a variety of useful patches. First, though, a bit of history...

The term 'sound synthesis' simply describes the process of using electronic circuitry to generate sound. Generally, synthesis involves the use of simple sound generators and modifiers, which, when combined in certain configurations, give rise to more complex and expressive sounds, sometimes aimed at emulating real instruments, but usually creating sounds of the type you'd never be able to get from an acoustic instrument. Today, the vast majority of new synthesisers built are software plug-ins from a huge range of developers – from bedroom coders to full-on corporations – but it's only just over a decade ago that the synth market was almost entirely centred on building hardware instruments and dominated by the venerable likes of Roland, Korg and Yamaha.

What many consider to be the world's first synthesiser came along in the 1876, when electrical engineer Elisha Gray invented the 'Musical Telegraph', a two-octave keyboard that contained the first sound generating electromagnetic oscillators – accidentally invented by Grey while he was working on telephone technology.

Circuit training

Over the next 50 years, engineers would continue to explore the use of electronic circuitry for musical applications, developing many of the building blocks of synthesis as we know it today, including the direct-current oscillator, noise generator and amplifier, as well as various forms of filter and modulator.

The earliest synthesisers were analogue, which meant that their sounds were generated

THS

In the third part of our series on the fundamentals of computer-based music production, we get you up to speed with software synthesisers and how to program them

by passing current through simple electronic circuits. These synths tended to be of 'subtractive' design, which is the architecture that still dominates today. A synth being subtractive simply means that it starts off with a harmonically rich sound source generated by one or more oscillators, such as a square wave, then filters specific frequencies out of it to achieve the desired result.

The new wave

In the 80s, a whole new wave of digital synths emerged – including the legendary likes of Yamaha's DX7, Roland's D50 and Korg's M1 and Wavestation – offering some new approaches to sound design, such as FM, additive and wavetable synthesis. These models didn't rely on discrete analogue circuits to generate sound, instead 'calculating' their sounds on dedicated CPUs and DSPs. This not only resulted in some very different sounds, but also meant that these synths could be much lighter and more reliable for live use than their analogue cousins.

"It's fair to say that soft synths have revolutionised the home studio"

With an increase in consumer computing power, the 90s saw digital synthesis taken further forward with more sophisticated ways to emulate instruments, including physical modelling and virtual analogue, and a rapid increase in the popularity and viability of software synthesisers.

It's fair to say that soft synths have revolutionised the home studio, opening the world of sophisticated, high-end synthesis up to the masses. Now, rather than filling entire rooms with cumbersome hardware synthesisers, we can house entire collections of virtual ones in our laptops – everything from meticulously modelled analogue emulations to outrageously powerful new designs that simply couldn't be realised in the real world.

Subtractive synthesis

Subtractive is the oldest and most ubiquitous of all the synthesis types, being easy to get to grips with and immediately satisfying, sonically speaking. It's based on the process of removing harmonics from a simple sound source to sculpt it into something more specific.

So, the first stage of any subtractive synth is its oscillators, which use a range of simple waveforms as starting points – commonly square, pulse, sawtooth and sine waves. Sometimes a noise oscillator is included as well, for generating white, pink and other colours of noise. Each of these waveforms has its own characteristic range and distribution of harmonics. For example, a square wave contains only odd number harmonics, giving it a distinctly hollow quality and making it an ideal starting point for 'wind' sounds.

The next stage is typically the filter, which 'filters out' harmonics above, below, or above *and* below, a certain frequency (the 'cutoff point'), or filtering out the cutoff frequency itself and its immediately adjacent harmonics. How sharply these harmonics are removed is determined by the filter design, with a 12dB/octave filter fading out frequencies at a rate of 12dB for every octave they go above or below the cutoff point, and a 24dB/octave filter producing a sharper cutoff of – you guessed it – 24dB per octave.

As a sawtooth wave contains a fairly even distribution of harmonics, from its fundamental frequency (the note it's playing) to well beyond the limits of human hearing, it's common when synthesising a string sound, for example, to use a low-pass filter – removing frequencies above the cutoff point – to soften the sound and produce a smoother tone.

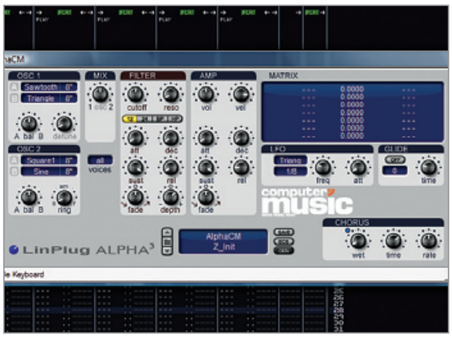
Move with the times

An important part of any subtractive synth is its ability to implement time-variant changes, aka 'modulation'. Quite simply, modulation is the automatic adjustment of a synth parameter over time by a modulator (such as an envelope generator or LFO). For example, a plucked acoustic instrument will typically start out producing a wide range of harmonics, but as the energy of the initial pluck dissipates, the higher frequencies will fade away first. To emulate this in a synth, a low-pass filter's cutoff point can be set to change over time by connecting it to an envelope generator within the synth's interface.

Modulation can be used to produce vibrato and tremolo by routing the relevant controls to a low frequency oscillator (or LFO – an oscillator that oscillates very slowly, used as a modulation source rather than a sound generator).

> Step by step

Programming a simple lead patch



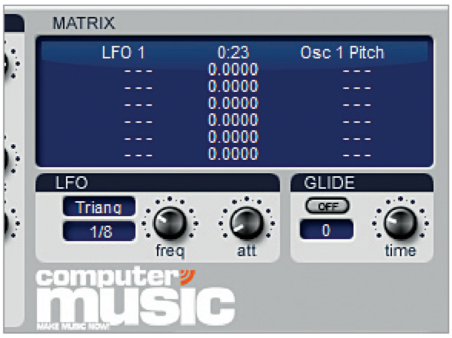
1 > For our first example, we'll use LinPlug's AlphaCM – a superb sounding, easy-to-program subtractive synth plug-in that you'll find in the **cm** Studio on our cover DVD. Get it installed if you haven't already, call it up in your VST-compatible DAW and either reset or 'initialise' the synth. Clicking the patch name brings up a presets menu. Scroll to the bottom of this and select **Z_Init**.

2 > For the basic saw wave lead sound we're going to make, we only need a single oscillator. The sawtooth wave is already selected in Oscillator 1 A, and to use this on its own we need to move the **Mix** dial all the way to the left, and the **Bal**ance dial in Osc 1 all the way to **A**.

3 > There you have it: a raw sawtooth sound. Not much to listen to, admittedly, but it's only the foundation of our sound. To give it a bit of character, head to the filter section. First, turn the **Cutoff** frequency down to about 10 o'clock and set the **Reso**nance (the amount by which the frequencies around the cutoff point are accentuated) to around 9 o'clock.

4 > Now let's add a little movement to the filter using the filter envelope. First, set the Filter **Depth** to between 2 and 3 o'clock, turn the **Sust**ain right down to 8 o'clock, and set the **Dec**ay to between 9 and 10 o'clock. This is the sort of classic enveloped filter sound used famously in Herbie Hancock's *Chameleon*.

5 > Next we can change the volume envelope. In the Amp section, move the **Decay** up to about 11 o'clock, and the **Sust**ain down to about 2 o'clock. This gives the sound a bit of an attack. It starts off at full volume, decays over the length of time set by Decay, then rests at the level set by Sustain.



6 > We can modulate the pitch of our saw wave, too. In the Matrix section, click the dashes in the top left slot and select **LFO 1** – this is your modulator. In the adjacent dashes on the right, select **Osc 1 Pitch** – this is the target. Now click the number in between them and move the value up or down to increase or decrease the depth of the modulation, which in this case creates a vibrato effect.

Oscillators – sound starts here

Oscillators are the basic building blocks of sound in a typical synthesiser. An oscillator produces a continuous, repetitive signal – such as a sine or square wave – which typically has its pitch assigned to track whatever notes are being played on your keyboard or input device.

An oscillator will usually offer a variety of selectable waveforms, each with its own characteristic distribution of frequency harmonics. A sine wave is a single frequency with no harmonics, while a square wave contains a theoretically infinite number of odd-number harmonics, which gives it a much thicker sound.

Many of today's soft synths offer a wider variety of waveforms than their hardware counterparts, ranging from more complex variants on the basic square, saw, triangle, sine, etc, to fully 'designed' samples. An increasing number even enable you to load your own samples or draw waveforms in manually.

Finger on the pulse

Oscillators can be modulated by LFOs, enabling the creation of tones that change over time. A pulse wave is similar to a square wave, only asymmetrical – more of a rectangle wave, if you will. The Pulse Width parameter (PW) determines how wide these pulses are and thus changes the harmonic content of the sound. Pulse Width can be modulated (Pulse Width Modulation, or PWM) by another, typically slower, oscillator, to give you a constantly shifting rectangular wave with a richer sound.

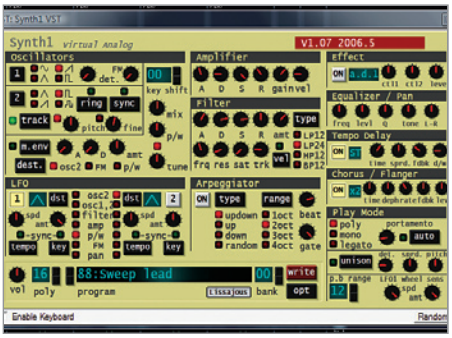
An oscillator's wave amplitude can also be used to modulate another oscillator's pitch. A simple application of this would be to create vibrato – the effect of pulsating the pitch of a

> "A sine wave is a single frequency with no harmonics"

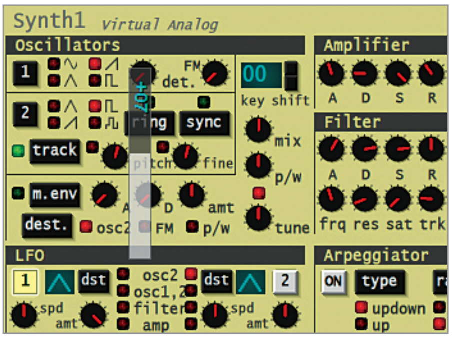
note up and down. But using one oscillator to modulate another – usually when both are in the audible range (ie, beyond typical LFO range) – can also lead to the creation of much more complex waves, which is the basic principle of Frequency Modulation (FM) synthesis.

Many synths also feature noise oscillators. White noise is a signal containing all of the frequencies within the audible range. It's useful for emulating typically non-pitched sounds, such as the smack of a snare or tick of a hi-hat.

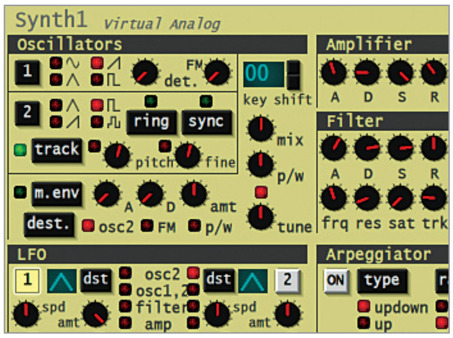
> Step by step Tuning oscillators



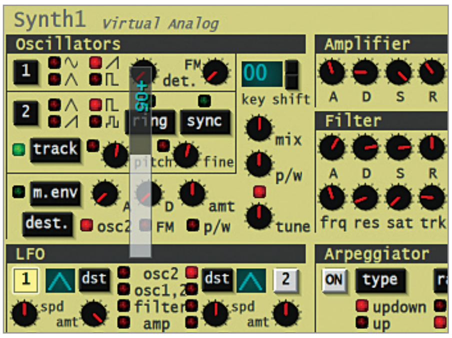
1 > For this example we're using Ichiro Toda's Synth1 (**www.geocities.jp/daichi1969/softsynth**) – a versatile and popular freeware subtractive synth with a similar architecture to Clavia's legendary Nord Lead hardware synth. To begin, let's load a preset: click the **Program** window and select preset **88: Sweep Lead**.



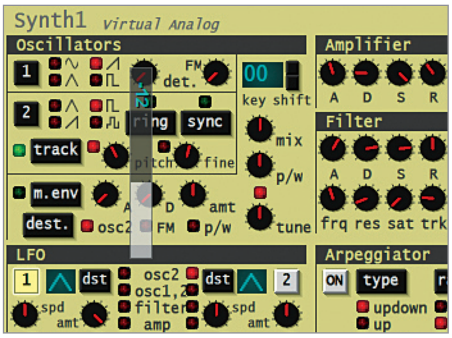
2 > This useful lead/pad sound deploys two oscillators, slightly detuned (ie, one is slightly out of tune with the other), running through a slowly envelope-swept filter. One of the first things we can try is changing the tuning of the oscillators. In the Oscillator 2 section, select and adjust the **Pitch** control to **+7**.



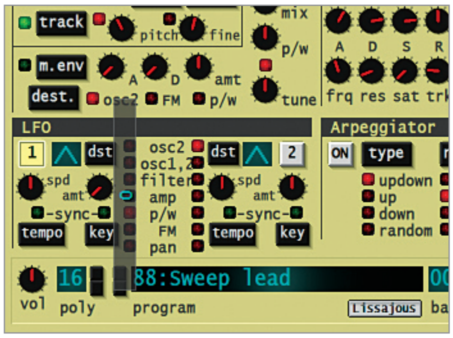
3 > You've now got oscillator 2 playing seven semitones higher than oscillator 1. This results in the classic open fifth sound, as if playing C and G together on a keyboard. An open fifth gives a thicker sound, due to the addition of extra harmonics.



4 > Another tuning we can try is a fourth. For this, select Oscillator 2's **Pitch** dial again and move the slider down to **+5**. The sounds will tend to get thicker as the tuning gap between the oscillators decreases, until they're in unison and mostly playing in phase.



5 > Another useful tuning is **-12**, with which Oscillator 2 plays an octave below Oscillator 1. This can add a lot of weight and low end to your patch, and many synths feature a dedicated sub-oscillator specifically for this purpose.



6 > You may have noticed that our patch has a shifting, uneasy feel to it. This is caused by the fairly heavy Pulse Width Modulation on Oscillator 2. To tone this down, go to LFO 1 and select the **Amount** dial. This is used to raise and lower the amount of modulation, which is PWM here. Taking it right down gives you a smoother but more clinical sound.

Filters – shaping frequencies

Filters are used to shape the frequency content of the signal generated by the oscillators.

A low-pass filter will remove all frequencies above the specified cutoff point. This makes it useful for reshaping harsh or buzzing sounds, for example, by taming and rolling off the top end to produce a smoother tone.

The strength with which a filter rolls off these frequencies is determined by the number of 'poles' it employs, each pole rolling off 6dB/octave. For example, a 2-pole low-pass filter will roll off frequencies at 12dB per octave, so if the filter's cutoff frequency was set at 440Hz, which is the note A4, the note A5 (880Hz) would be 12dB quieter than A4, since it's exactly an octave higher. In exactly the same way, a 4-pole low-pass filter rolls off the frequencies at a much sharper rate of 24dB/octave.

A high-pass filter will remove all frequencies below the cutoff point. This can be used to remove unwanted low-frequency weight or rumble from a sound, and also to reshape a signal in order to create top-heavy sounds, such as bells, 'glassy' effects, hi-hats, cymbals, etc.

A band-pass filter can be thought of as a combined low-pass and high-pass filter, since it only lets frequencies within a certain range of the cutoff point through, rolling off the frequencies both above and below it. These are a good choice when you want to focus on a very specific frequency band in a sound, such as when preventing thin-sounding lead parts from fighting for space with other elements in a mix.

Resonant frequencies

Resonance is the amplification of frequencies near the cutoff point, which can be useful in synth sound design. With the resonance at zero, frequencies are simply rolled off directly from the cutoff point. Turn the resonance up, though, and a narrow band of frequencies around the cutoff point is boosted, producing an increasingly sharp frequency spike in the sound, according to how high the parameter is set.

In some analogue filters, both real and virtual, the resonance circuit is designed so that the filter will start to self-oscillate if it's turned up high enough. This simply means the filter starts producing a sine wave tone of its own. If the cutoff point is set to track the notes played on the synth, the self-oscillating filter can act much like an additional oscillator.

Another filter type worth a mention is the notch filter, which, in the opposite way to how a band-pass filter works, removes frequencies at and immediately around the cutoff point, leaving everything above and below unaltered.

> Step by step
Creating a plucked sound

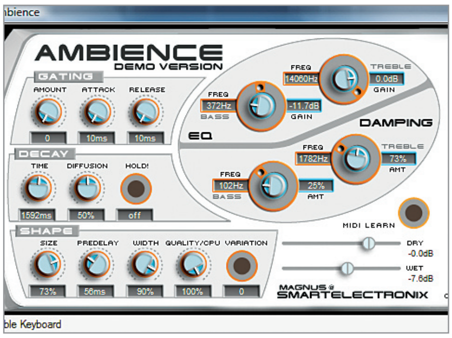
1 > We're using Linplug AlphaCM again for this example. As we did earlier, initialise the patch by selecting the **Z_Init** preset. We're only going to use the one oscillator, so move the **Mix** knob all the way to **1**.

2 > To create the basis of a soft, plucked sound, we'll use a sine wave and a triangle wave mixed together. Click the wave display in Oscillator 1 A to bring up the wave menu and select **Sine**. We've already got a Triangle wave in slot B.

3 > To create the actual plucking effect, we need to turn to the filter envelope. First, bring the **Cutoff** down to 9 o'clock and turn the **Depth** all the way up to 4 o'clock. Bring the filter envelope **Sustain** right down to 8 o'clock, and the **Decay** up to between 9 and 10 o'clock.

4 > This gives us a short 'thrip' sound as the filter goes from open to closed very quickly after a note is played. Next, copy this filter envelope shape to the amp envelope, which gives the sound the particular plucked quality we're after. Bring the amp envelope **Sustain** down to 8 o'clock and the **Decay** up to 10 o'clock.

5 > Now we've got a simple plucked sound. Let's add some depth and width to it with AlphaCM's built-in Chorus effect. Bring the Chorus' **Wet** dial up to about 10 or 11 o'clock. Experiment with the **Time** and **Rate** parameters to control the colour and intensity of the chorus.



6 > Our plucked sound is still unmistakably synthetic. To make it sound more natural, all we need to do is add a bit of reverb. Here we're using Smartelectronix's excellent Ambience plug-in (**magnus.smartelectronix.com**) set to the **Amoeba Hall** preset. Very nice.

Envelopes and LFOs

One of the most important aspects of synth programming is modulation. This can take the form of anything from changing the volume of a sound over time after a note is struck, to altering filter and tuning settings in order to create sounds that evolve and transform over time of their own accord.

An LFO (low frequency oscillator) is much like any other oscillator, but designed to oscillate at very low, subsonic frequencies, where, rather than being heard, it can be used to modulate various synthesiser parameters.

LFOs were originally used to emulate vibrato (modulating the pitch of a note) and tremolo (modulating the volume). Today they're used for all manner of things – subtly modulating the pitch of one oscillator while another stays static, providing a sense of movement and depth in pads or strings, for example.

Pushing the envelope

Another essential part of synthesis is envelopes, which describe what happens to a certain parameter over time after a note is struck. For example, a bass drum has a loud attack at the beginning of its sound, followed by a rapid drop in volume after the initial burst, then a smooth decay to silence. An amplitude envelope can be set up to emulate this, instructing the amplifier to let the sound through straight after being triggered, then decay smoothly and quickly.

The most common type of envelope is known as an ADSR, standing for Attack, Decay, Sustain and Release. Attack determines the length of time it takes for the sound to reach full volume after it starts – an instantaneous attack would be used on a sharp sound, such as a percussion instrument, while a longer attack fades a sound in smoothly, and would be used when programming, say, a string sound.

Decay refers to the length of time it takes for the sound to go from its full volume level – set by the Attack – to the volume level set by the Sustain. Sustain sets the volume the sound will play at while the note triggering it is held down. 100% Sustain would keep the sound playing at full volume for as long as the note was held down, while at 0% Sustain, your sound would reach full volume following the Attack, then fade to silence over whatever length of time your Decay was set to, even with the note held down.

Finally, the Release segment dictates the length of time it takes your sound to go from its current volume to silence once the key is released. A zero Release means the sound will stop as soon as you lift your finger from the key, while a long Release will let the sound fade out smoothly over a period of time.

> Step by step

Constructing a pad sound

1 > Pad sounds are generally defined by long, slow envelopes and a sense of movement created through subtle detuning and modulation. Load AlphaCM's **Z_Init** preset again to initialise the synth, and move the A/B **Bal**ance dials in both the oscillator sections to **A**.

2 > We're going to use saw and square waves in this example, so click the wave name display in oscillator 1's A section and select **Sawtooth**, then do the same in the A section of oscillator 2 but select **Square2** instead. Set oscillator 1's **Detune** to 9 o'clock.

3 > The filter section is where we can build a convincing sense of movement into the sound. Set the **Cutoff** all the way down to 8 o'clock and the **Reso**nance to about 12 o'clock. Select **HP** from the mode menu below to set it up as a high-pass filter.

4 > On to the filter envelope: set the **Dec**ay all the way up to 4 o'clock, the **Sust**ain down to 8 o'clock, and the **Rel**ease at 3 o'clock. Set the filter **Depth** to between 3 and 4 o'clock. With the envelope set up like this, the Cutoff will be modulated to slowly move down as the note is held, then continue dropping smoothly once it's released.

5 > For the Amp envelope, set the **Att**ack at about 10 o'clock and the **Rel**ease at about 1 o'clock. This will make our sound fade in smoothly when a note is played, then fade out smoothly when it's released. Let's also add some chorus – set the **Wet** dial to around 1 or 2 o'clock.

6 > Detuning the oscillators is a great way to add thickness, richness and depth to a pad sound. This can be done in AlphaCM's modulation Matrix. Click the left slot of an empty row and select **Constant**, then click the right-most slot of the same row and select **Osc 2 Pitch**. Set the value in between to **7.00** for an open fifth tuning.

FM synthesis

Frequency Modulation (FM) synthesis is based on the principle of modulating one simple oscillator waveform's pitch with the movement of another oscillator in order to produce rich, complex sounds.

The process starts with an audible oscillator (the 'carrier'), the pitch of which is modulated by the amplitude of a second oscillator (the 'modulator'), typically at a rate that would also put it in the audible range. The louder the modulator signal, the more the carrier signal's pitch changes – the amount by which the carrier signal is modulated is referred to as the 'deviation'.

FM synthesis is well suited to producing both harmonic and inharmonic sounds. When the carrier and modulator frequencies are harmonically related, a wide variety of synthetic and instrument-like timbres can be generated – when the modulator signal is not harmonically related to the carrier signal, dissonant sounds, such as bell-like tones, are produced.

> "FM synthesis is well suited to producing both harmonic and inharmonic sounds"

The oscillators in FM synths are referred to as 'operators'. An FM synthesiser will usually have either four or eight operators onboard, with a variety of preset configurations available to determine how they're routed – it's possible to connect operators serially, in long chains, or have them playing in parallel. It's also possible to route an Operator back into itself, making it feed back and modulate its own frequency.

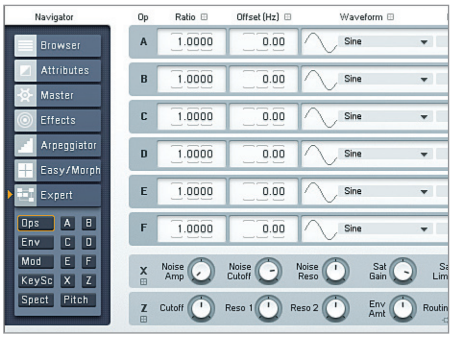
FM jargon

Oscillator pitches in FM synth parlance are described as 'ratios', with an Operator set to a ratio of 0.5 oscillating at half the frequency of an Operator set to a ratio of 1, and an Operator set to a ratio of 2 oscillating at double. With each doubling of its ratio, the pitch of an Operator goes up an octave.

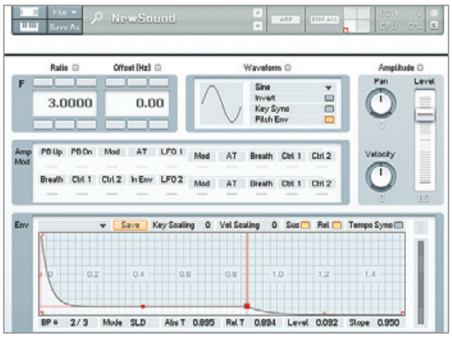
Beyond that, FM synths tend to feature many of the same components as analogue synths, such as envelope generators, which can be assigned to control the volume and pitch of an Operator, and LFOs, which really act as simple, low-frequency Operators for adding effects such a vibrato.

> Step by step

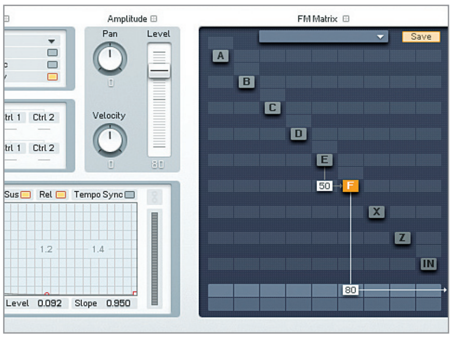
Making an FM bell tone



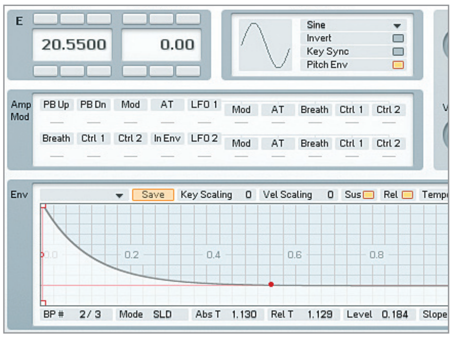
1 > For this simple FM demonstration we're using Native Instruments' FM8. Bell-like tones are produced when one oscillator modulates the pitch of another and there's no harmonic relationship between the two. Enter **Expert** mode and notice that we have a single operator (F) in the FM Matrix on the right-hand side of the interface. This is our carrier.



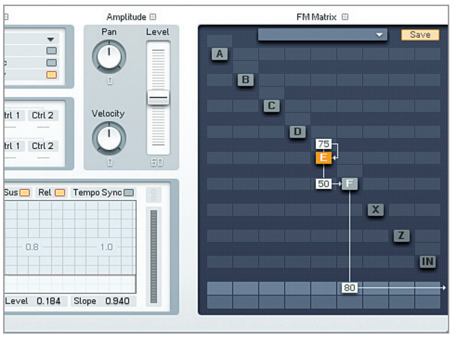
2 > Currently we're playing a simple sine wave with this patch. Select F in the FM Matrix to bring up this operator's controls. Up the **Ratio** to **3.0000** and you should hear that the operator is now playing two octaves higher than before. Add a volume envelope, too.



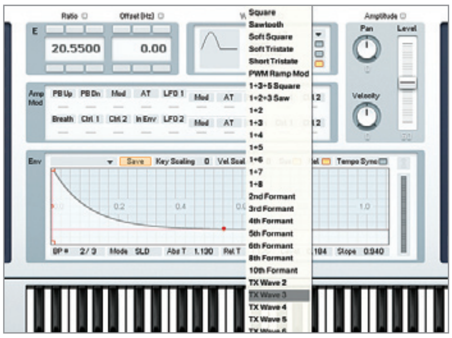
3 > The next step is to modulate the pitch of this operator with a second operator. Locate the box in the FM Matrix at the point where the E column and F row intersect and click to activate it. Now move the mouse up to feed operator E's signal into operator F – set the value to about **50**. Operator E is now our modulator.



4 > Right-click **E** in the FM Matrix to activate it, then left-click it to bring up the controls and turn the **Ratio** up to **20.5500**. We've transformed a simple sine wave into a dissonant-sounding bell tone. We can control how this operator changes the sound over time by setting an envelope here.



5 > We can add some higher harmonics to our bell sound by setting operator E up to modulate itself as well as operator F. To do this, select the box in the FM Matrix directly above E and drag the value up as before to about **75**.



6 > We can manipulate this sound further by changing the waveforms used both by the carrier and modulator. A softer, more hollow sounding bell tone can be created by selecting **TX Wave 3** for operator E. This is located in the menu in the **Waveform** section.

Sample-based synthesis

Sample-based synthesis involves the use of recorded samples, rather than analogue oscillators, to more accurately reproduce the tonal characteristics and playing nuances of real instruments, or create the sorts of sounds that analogue oscillators aren't capable of.

In early sample-based synths, memory was a scarce resource, with a whole synthesiser's sound library having to be squeezed into just a megabyte or two. This meant the samples themselves had to be extremely short – often just a single wave cycle from a recorded instrument that was looped to provide a continuous tone – and that most of the sound shaping and performance characteristics had to be handled using cleverly programmed filter and volume envelopes.

The realism of a sample-based synth patch is largely determined by the size and quality of the sample library available to it. While slowing down and speeding up a short instrument sample will enable it to be played back melodically on a keyboard, the quality suffers dramatically when it's pitched too far up or down. To overcome this, multiple samples of the instrument being played at various pitches are recorded and laid out across the keyboard range – this is known as multisampling.

Multisampling can also be employed to add a level of performance realism. Multiple samples can be layered up within a patch and triggered according to various performance parameters. For example, a loud piano note sample could be layered on top of a quieter one, with the velocity of an incoming MIDI note determining which one would be triggered. By combining a few short samples in this way, moderately realistic sample-based instruments could be created.

Keyed up

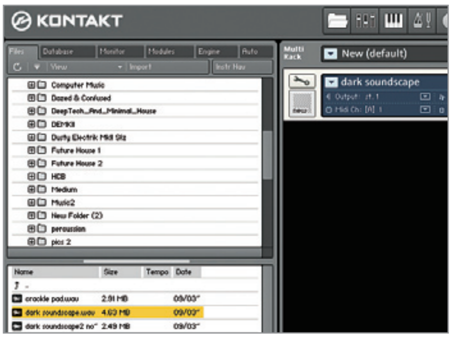
Today's sample-based synths, aka ROMplers, come with massive sample libraries, facilitating incredible levels of realism. A grand piano, say, can be recreated in meticulous detail by using long, high quality samples of every note of the original instrument played at different velocities.

Most commercially available soft samplers now include huge libraries of multisampled instruments, as well as the ability to import sounds of your own making/choosing. Samplers like Kontakt and HALion offer many of the same features found on powerful subtractive synths, such as multiple filters, complex modulation and routing options, and high-quality effects.

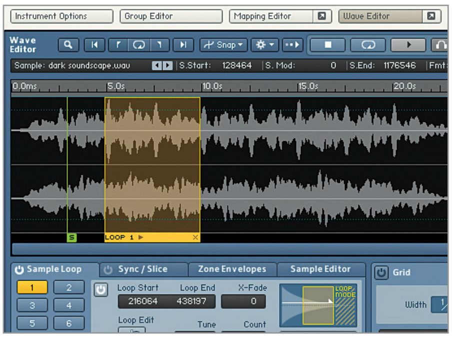
Sample-based synthesis is even used to recreate vintage synthesisers, and can also be utilised in more imaginative ways – ie, turning a single human voice into a choir pad.

> Step by step

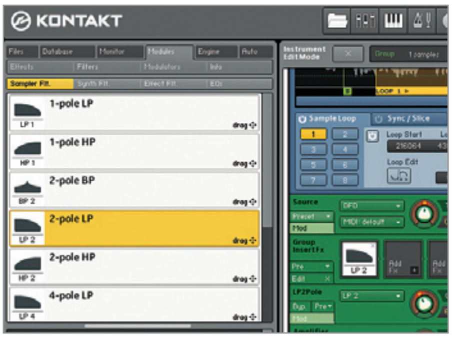
Creating a sampled vocal pad



1 > Samplers like NI's Kontakt combine sample playback with powerful synthesis functions, offering an extraordinary range of processing and routing options. Let's take a vocal sample and turn it into a pad sound. The first thing to do is go to the **File** menu and import a suitable sample, such as a choir or solo singer sustaining a note.

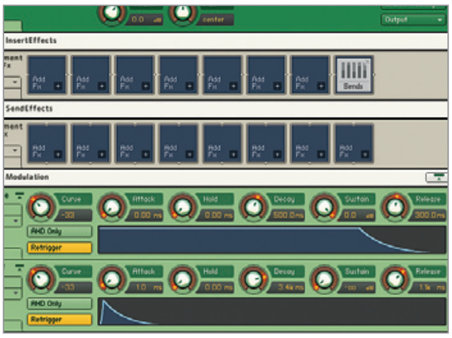


2 > This should bring up a new module in Kontakt's instrument window. Click the **Tool** icon on the module and select **Wave Editor**. Here you can set the sample start and end points, as well as make the sample loop smoothly by clicking the switch next to Loop Start and adjusting the **Loop Start** and **Loop End** points.



3 > The first processor we'll bring in is a filter. Go to the **Modules** window on the left and select Filters. We'll use the **2-pole LP** filter here, so drag it from its window into an empty **Add FX** slot in the Group Insert FX box underneath the Wave Editor, as shown above.

4 > We can lower the cutoff and bring some resonance in here. To apply an envelope to the filter, select Modulators in the Modules window and drag an **AHDSR** module from the Envelopes section onto the **Cutoff** dial just underneath Group Insert FX.



5 > If we scroll to the bottom of the Instrument window now, we'll find our envelope modules. Our filter envelope is right at the bottom, labelled **Cutoff**. Take the **Sustain** right down and bring the **Decay** up to **3.4s**.

6 > We can also select Effects from the Modules window, so add a **Chorus** to a spare **Add FX** slot in the Insert Instrument FX rack. The chorus controls should appear directly below this, offering **Depth**, **Speed** and **Phase** parameters.

Physical modelling

Prior to physical modelling, synthesisers aimed to approximate the sound of the particular instruments they were trying to emulate using either oscillators, filtering and modulation, or samples. Physical modelling, however, does away with this by generating sounds using mathematical models of the actual physical components of each instrument. For example, physical models can be made by calculating what happens when a bow is dragged across a string, or when a stick strikes a drumhead.

The main advantage of this method is the level of flexibility that physical models enable. While sample-based synthesis is limited to crudely emulating playing characteristics using volume or filter envelopes – or relying on vast sample libraries to get around this – with a physical model, these parameters can be emulated and assigned to real-time controls. Such controls could be anything from the force used to pluck a string, to the width of a violin bow or the damping of a drumhead.

The concept of physical modelling has been around since the early 70s, but due to the extreme demands it makes on the processor doing the modelling, it wasn't seen as practical for real-time work until the more efficient digital waveguide synthesis emerged in the late 80s.

Model behaviour

Akin to its physical brother, analogue modelling (also known as 'virtual analogue' or 'VA'), on which a great many soft synths are based, first appeared in Clavia's Nord Lead synth in 1995.

Digital synthesis became popular in the 80s thanks to instruments like the Yamaha DX7 and Korg M1, which could emulate acoustic instruments with greater realism and fewer tuning problems than their analogue cousins.

However, the tide turned towards the end of the 80s, driven by the growing popularity of underground dance music, and the warmth and instability of classic analogue synthesisers started to become fashionable again. And while digital subtractive synths, such as the Roland JD-800, far exceeded the functionality and reliability of true analogue synths, their sound was seen as somewhat sterile.

Things came full circle in the early 90s when synth manufacturers began developing digital techniques to model the characteristics of the electronic circuitry behind analogue oscillators, amps and filters. Ironically, one of the primary goals of analogue modelling is to emulate the inherent instability and non-linear distortions that early analogue synths produced, leading to a more colourful and 'organic' sound.

> Step by step Modelling a Fender Rhodes

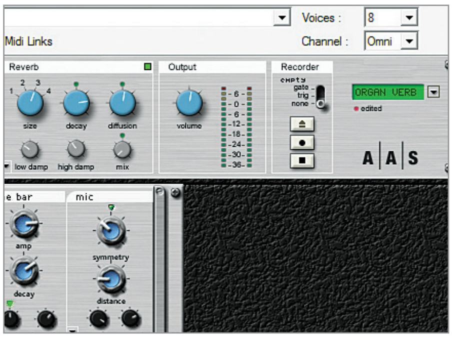
1 > For this example, we're using the amazing Tassman 4, a modular physical modelling synth from Applied Acoustics Systems (**www.applied-acoustics.com**). We're going to make a Rhodes-like electric piano sound, so the first thing to do is load the **Mid** patch from the **E Piano1** directory.

2 > Much of the timbre of this electric piano patch is controlled in the Row 1 rack, which hosts various parameters relating to physical components of the instrument, such as the tone bar and mic position. To soften the sound a little, move the **Stiffness** and **Strength** dials, under Hammer, to 12 o'clock.

3 > We can add a bit of a mechanical clunk to the attack of our piano sound with the Tine controls. Move the Tine **Amp** up to 3 o'clock – you should hear the sound take on a harder, more aggressive attack. Bring this down to a short click by turning the **Decay** control down to 9 o'clock.

4 > We can give the sound a bit more bite in the Tone Bar and Mic sections. Turn the Tone Bar **Amp** up to 3 o'clock and bring the Mic **Distance** up to 9 o'clock. We can control the smoothness of the sound now using the **Ampin** dial in the Mic section.

5 > We can add an authentic modulating volume effect to our sound now using the Tremolo section in Row 2. Use the switch to turn the Tremolo **On**. The **Depth** setting determines how strong the effect is, while **Speed** configures the tremolo rate.



6 > The Output rack, at the top, is the last stage our sound goes through. Load the **Organ Verb** from the menu over on the right. Now increase the **Decay** in the Reverb section to 3 o'clock and engage the delay by clicking the green button at the top right of the Delay section.

Modular synthesis

While most analogue synths have a fixed architecture and signal path – with oscillators, filters and various modulators and effects laid out in a scheme that can't be changed, thus feeding into each other in a specific way – a modular synth enables you to arrange and route its component parts any way you like.

Until Moog and EMS changed everything with the Minimoog and VCS-3 respectively at the start of the 70s, synths were massive modular beasts, with which oscillators, filters and simple sequencer modules were patched together in various configurations by physically plugging them into each other. Because of this, the patching process itself was an intrinsic part of the sound design process. For example, if you wanted to synthesise a simple percussion instrument, you might start by patching an envelope generator into the pitch control of a simple sine wave oscillator, which would in turn be patched into an amplifier controlled by another envelope generator, giving you a simple sine wave sound that can have its pitch and volume adjusted to make short thuds, clicks or tom-like sounds.

If you wanted to turn this into a more atonal, FM-like sound, such as a gong or bell, you could simply add another oscillator and have that control the pitch of the sine wave instead, or add a mixer module and sum the control signals from both the envelope generator and the additional oscillator, or figure out any number of other ways to achieve a similar effect depending on the level of control you require.

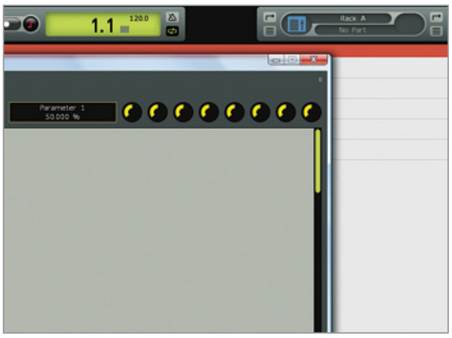
Synth scientist

Modular synths are also well suited to 'pure' synthesis, as opposed to instrument emulation. With a modular synth, you're able to essentially copy and expand on the architecture of any synth ever designed – providing you have the necessary modules. With hardware modular synths, modules would typically be monophonic – meaning an oscillator could only provide a single tonal output at any one time, and a filter could only filter a single signal. In today's software modular synths, this isn't usually the case, and a single module will be polyphonic (able to play more than one note at once).

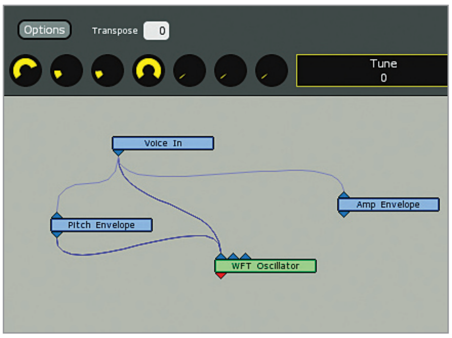
Modular soft synths are now popular among experimental musicians requiring a greater degree of scope and flexibility in sound design than fixed-path synths can offer, as well as those with very specific compositional and performance needs, since they enable sophisticated, custom instruments and sequencing systems to be designed intuitively.

> Step by step

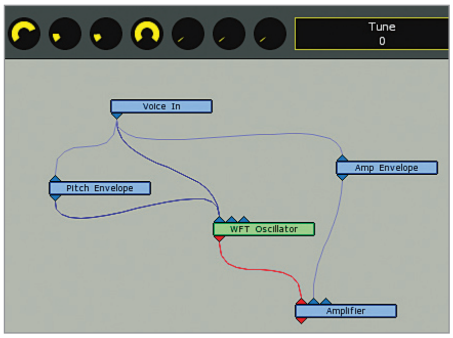
Making a kick drum with a modular synth



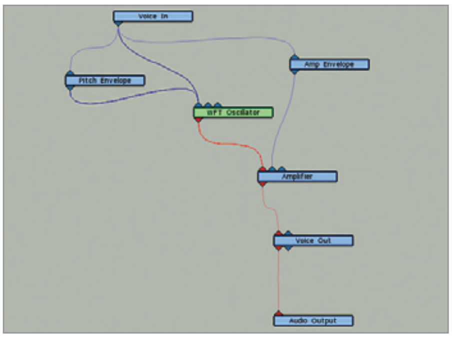
1 > In this example, we'll be using the free MuSynth in the equally free Mu.Lab DAW (**www.mutools.com**). First, start a new song and switch to the **Rack Desk** window. Click in the empty top slot on Rack A and select **Synths»MuSynth**. Click **Options** in the MuSynth window and select **Open Deep Editor**.



2 > To add modules to the synth, right-click and select **Add Plugin**. We need a way to tell the synth when keyboard keys are being pressed, so add a **Voice In** module. Then add two **ADSR Envelopes** and a **WFT Oscillator**. Connect these to our Voice In element by clicking and dragging between the connector points, as shown.

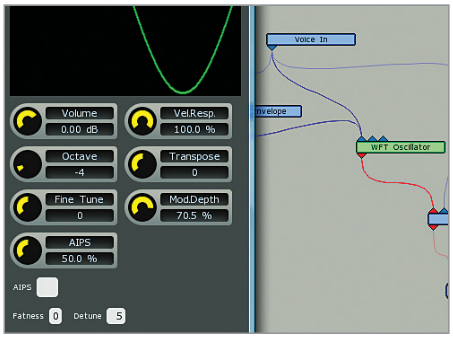


3 > Now, each incoming note will trigger the elements we just set up. However, as yet, they won't actually make any noise. For that, we first need to add an **Amplifier** – do that now. Connect the output of the first envelope to the Control input of the WFT Oscillator, and connect the second envelope to the Amplifier.



4 > Lastly, we need **Voice Out** and **Audio Output** elements. Connect these as shown. Now we have the architecture of our bass drum synthesiser laid out. The element parameters can be edited by right-clicking on the element you want to change and selecting **Edit**.

5 > Edit the first envelope, which is controlling the pitch of the oscillator. Set the **Attack Time** to **0ms**, the **Decay Time** to **182ms**, the **Attack Level** to **100%** and the **Sustain** to **0%**. Edit the second envelope, which is controlling the amp: **Attack Time 1ms**, **Decay Time 289ms**, **Sustain 0%**.



6 > Now we just need to set up our oscillator. Set the **Octave** to **-4** and the **Mod Depth** to **70%**. You can change the oscillator waveform by right-clicking the wave in the window. You should now have a simple bass drum synthesiser that responds to MIDI notes from your keyboard.

Synthesis tips and tricks

THINK BEFORE YOU ACT

Even the most basic subtractive synth can be a deceptively involved instrument to program. Synthesisers welcome experimentation, but very often it can be helpful to take a step back and consider how a sound you want to program would be constructed before you actually start doing it.

As a square wave has a hollow tone, it makes a good starting point for synthesising woodwind-like instruments, for example, whereas a saw wave has a much smoother tone, which makes it particularly good for emulating string-like instruments. Another thing to consider might be how the tone of the instrument changes over time, and how you can make that happen with the filter and filter envelope sections.

REACH NEW DEPTHS

Acoustic instruments and sounds in nature aren't static things – they tend to shift, dance around and evolve continually. If you loop a single wave from a recording of a human voice, a string instrument, or even an analogue synth, you tend to get an unnaturally colder, synthetic-sounding result, which may or may not be what you're looking for.

It's always worth thinking about ways in which you can add subtle, natural movement to your synthesised sounds. One of the most effective it is to set your synth's LFOs up to modulate parameters like oscillator pitch, filter cutoff and pulse width modulation.

ALL ABOUT THE FINE-DETUNING

Another good way to add depth and richness to your synth patches is to use multiple oscillators slightly detuned against each other. If you were making a two-oscillator string sound, say, you might choose to detune one oscillator up 5 cents, and the other down 5 cents.

For even thicker sounds – especially big trance-style synth chords – you could use more oscillators and a greater range of detuning. Some synths offer a unison mode, which multiplies the number of voices your synth uses for each note and enables you to choose the degree of detuning.

Don't settle for a static sound – use LFOs to modulate all manner of parameters and add a sense of movement

STRIKE A CHORD

Coarse detuning can be used to add even more thickness to your synth sounds, as well as creating interesting melodic effects.

The simplest way to add weight to a synth sound is to add an oscillator and tune it down an octave (-12 semitones). Many synths also feature a dedicated sub-oscillator for adding a simple tone tuned, typically, an octave or two down.

Tuning an oscillator up +7 semitones gives the classic open fifth sound. This can be very effective with lead sounds and pads, particularly when playing big chords.

EXPRESS YOURSELF

You can make your patches much more expressive and organic by assigning synth parameters to the various MIDI controllers sent out by your keyboard.

Vangelis' legendary *Blade Runner* and *Chariots of Fire* soundtracks featured some highly expressive synth sounds, courtesy of the Yamaha CS80. That synth was famous for its polyphonic aftertouch, which gave the player very expressive fingertip control over parameters like cutoff frequency on sustained notes and chords. By assigning filter, envelope, oscillator and effects parameters to note velocity, aftertouch and any other controls that your keyboard outputs, you can create patches that morph and evolve with your playing.

LAYER UP

To make your leads, basses and chords really stand out, try layering multiple synth parts on top of each other to form much larger, multi-voice sounds. This can usually be done by setting a number of MIDI tracks in your sequencer to the same MIDI channel, then a plug-in synth into each one. It's a common trick in genres like drum 'n' bass and dubstep, where the bass needs to carry the tune. Layering a distorted, mid-range bass sound or two with a deep, low sub-bass from another synth can help you create huge bass sounds that fill out a broad range of frequencies.

TAKE CONTROL

Sometimes the best way to program a soft synth is to ignore the screen and just play around with the dials. Of course, this isn't really an option when you're relying on a mouse or touchpad, so it's a good idea to invest in a hardware MIDI controller, then assign your synth's various parameters to its rotary dials, faders and buttons.

This also makes it much easier to use the synth as a performance instrument, on which you can manipulate filter or effects settings in real-time to increase the expressiveness of your playing.

DON'T NEGLECT EFFECTS

Effects processes are an important part of synthesis. Adding a chorus will not only widen a sound in the stereo field, but can turn a simple string sound into a whole string section. Distortion can be used subtly, to add additional harmonics, energy or growl to the upper mid-range of a sound, or less subtly, emulating a guitar being overdriven into a guitar amp, for example.

Modular synthesisers expand the scope of what you can do with effects by enabling you to integrate them more deeply into the architecture of your patches. For example, adding distortion to an oscillator before it's sent to a filter is a classic technique for producing big synth leads. cm

Why use one synth when you can use three? This technique is often used for the thick sounds in dubstep